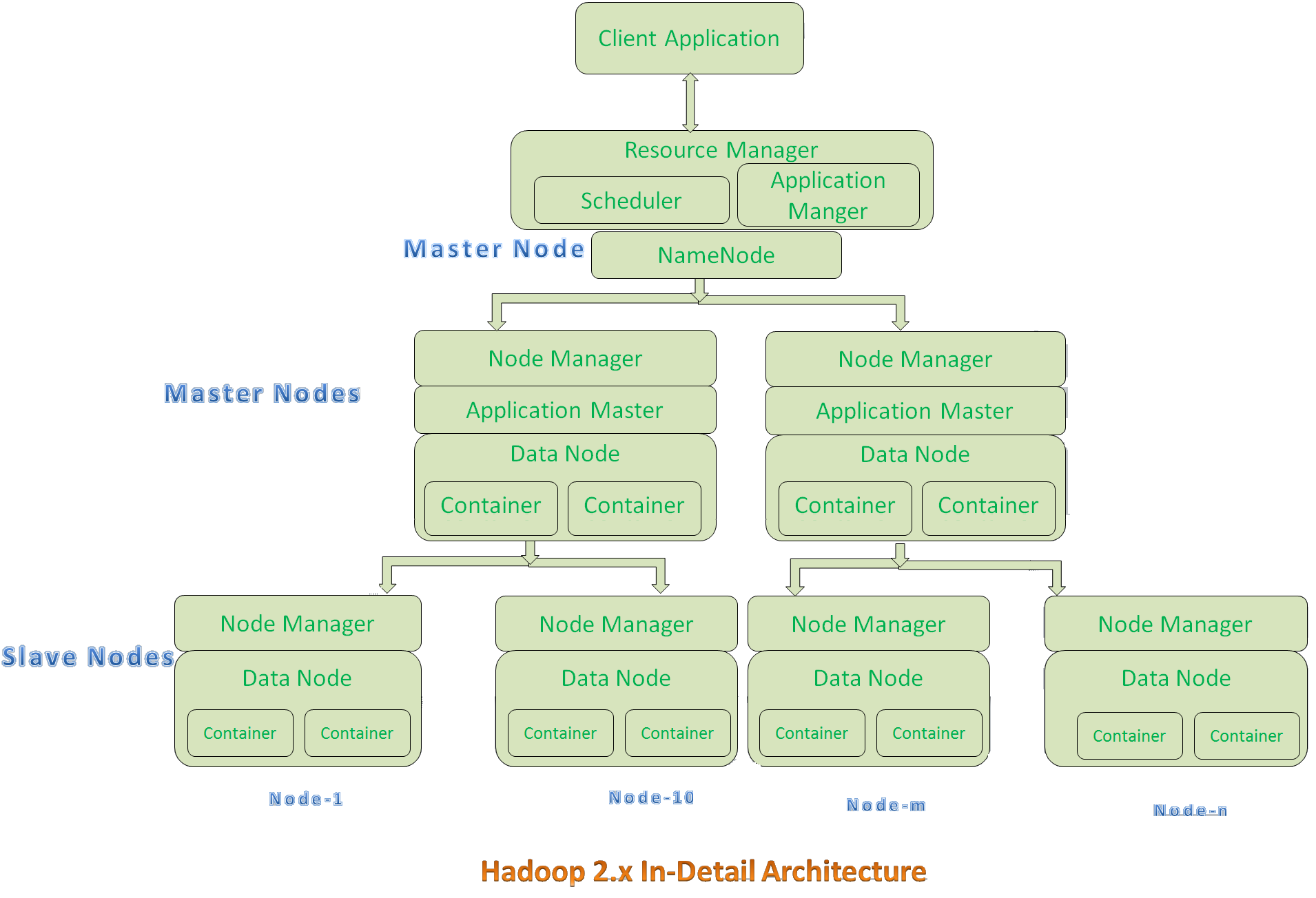
**Components of Hadoop 2.x**

* HDFS
* YARN
* MapReduce

These three are also known as Three Pillars of Hadoop 2

****

* All Master Nodes and Slave Nodes contains both MapReduce and HDFS Components.  
  Each Master Node has two components:
  1. Resource Manager
  2. HDFS

Its HDFS component is also known as NameNode, used to store Meta Data.  
In Hadoop 2.x, some more Nodes acts as Master Nodes as shown in the above diagram. Each this 2nd level Master Node has 3 components:

* + Node Manager
  + Application Master
  + Data Node
* Each this 2nd level Master Node again contains one or more Slave Nodes as shown in the above diagram.
* These Slave Nodes have two components:
  1. Node Manager
  2. HDFS

Its HDFS component is called as Data Node, used to store actual data of our application Data.

***HDFS:***

* HDFS stands for Hadoop Distributed File System. It is used as a Distributed Storage System in Hadoop Architecture.
* HDFS is the Hadoop file system and comprises two major components:

namespaces and blocks storage service.

* The namespace service manages operations on files and directories, such as creating and modifying files and directories. The block storage service implements data node cluster management, block operations and replications.
* **Hadoop Distributed File System** is a block-structured file system where each file is divided into blocks of a pre-determined size. These blocks are stored across a cluster of one or several machines.
* The default size of each block is 128 MB in Apache Hadoop 2.x which you can configure as per your requirement.

***YARN:***

* YARN is often called the operating system of Hadoop because it is responsible for managing and monitoring workloads, maintaining a multi-tenant environment, implementing security controls, and managing high availability features of Hadoop.
* YARN is designed to allow multiple, diverse user applications to run on a multi-tenant platform.
* One of the most significant benefits of YARN is that we are no longer limited to working the often I/O intensive, high latency MapReduce framework.

***MapReduce:***

* The MapReduce algorithm contains two important tasks, namely Mapping and Reduce.
* Mapping takes a set of data and converts it into another set of data, where individual elements are broken down into tuples (key/value pairs).
* Secondly reduce, which takes the output from a map as an input and combines those data tuples into a smaller set of tuples.
* With MapReduce it is easy to scale data processing over multiple computing nodes.
* During a MapReduce job, Hadoop sends the Map and Reduce tasks to the appropriate servers in the cluster.
* The framework manages all the details of data-passing such as issuing tasks, verifying task completion, and copying data around the cluster between the nodes.